REMARKS

This is a full and timely response to the final Office Action of September 22, 2005. Reexamination, reconsideration, and allowance of the application and all presently pending claims are respectfully requested.

Upon entry of this Amendment, claims 1, 2, 4-8, 10-16, 18-27, and 30 are pending in this application. Claims 1, 7, 10, 11, 12, 16, 18-22, and 30 are directly amended herein, and claims 3, 9, 17, 28, and 29 are canceled. It is believed that the foregoing amendments add no new matter to the present application.

Claim Objections

Claims 18 and 19 are objected to in the outstanding office action due to alleged informalities. In particular, the Office Action notes an irregular trend in the dependency of claims 18 and 19 on their respective independent claim. Claims 18 and 19 have been amended herein to depend from claim 16. Accordingly, Applicant respectfully requests that the objections to claims 18 and 19 be withdrawn.

Response to §112 Rejections

Claims 1, 2, 4-6, 12-15, 18, 19, 21, and 25-27 presently stand rejected under 35 U.S.C. §112, first and second paragraphs, as failing to comply with the written description requirement and failing to comply with the enablement requirement, respectively. Claims 1, 12, and 21 have been amended herein thereby mooting these rejections. Accordingly, Applicant respectfully requests that the 35 U.S.C. §112 rejections of claims 1, 2, 4-6, 12-15, 18, 19, 21, and 25-27 be withdrawn.

Response to §103 Rejections

In order for a claim to be properly rejected under 35 U.S.C. §103, the combined teachings of the prior art references must suggest all features of the claimed invention to one of ordinary skill in the art. See, *e.g.*, *In Re Dow Chemical*, 5 U.S.P.Q.2d 1529, 1531 (Fed. Cir. 1988), and *In re Keller*, 208 U.S.P.Q. 871, 881 (C.C.P.A. 1981). In addition, "(t)he PTO has the burden under section 103 to establish a *prima facie* case of obviousness. It can satisfy this burden only by showing some objective teaching in the prior art or that knowledge generally available to one of ordinary skill in the art would lead that individual to combine the relevant teachings of the references." *In re Fine*, 837 F.2d 1071, 5 U.S.P.Q.2d 1596, 1598 (Fed. Cir. 1988) (Citations omitted). Furthermore, the Federal Circuit has stated that "(i)t is impermissible, however, to simply engage in hindsight reconstruction of the claimed invention, using the applicant's structure as a template and selecting elements from references to fill the gaps." *In re Gorman*, 933 F.2d 982, 987, 18 U.S.P.Q.2d 1885 (1991).

Claim 1

Claim 1 presently stands rejected under 35 U.S.C. §103 as unpatentable over *Jensen* (U.S. Patent No. 6,587,937) in view of *Burton* (U.S. Patent No. 6,738,865) and in further view of Free On-Line Dictionary of Computing's "Pipeline" (hereinafter FOLDOC). Claim 1 reads as follows:

> 1. A computer system for efficiently executing instructions of computer programs, comprising:
>
> processing circuitry having a pipeline, said pipeline configured to execute instructions from one of a plurality of programs, said processing circuitry configured to stop executing said one program during a first context switch in response to a first context switch command and to resume executing said one program during a second context switch in response to a second context switch command;
>
> cache memory;
>
> computer memory having a plurality of addresses; and
>
> ***memory control circuitry coupled to said processing circuitry, said memory control circuitry configured to store, in response to said***

first context switch command, in computer memory, data written by said pipeline during execution of said one program and to store an indicator indicative of whether said data was accessed by the processing circuitry during a particular time period prior to said first context switch in executing said instructions from said one program, said memory control circuitry, in response to said second context switch command, configured to identify one of said addresses of said computer memory that is storing said indicator corresponding to said data previously written by said pipeline during execution of an instruction of said one computer program prior to said first context switch, said memory control circuitry further configured to make a determination, based on said indicator, whether said data was accessed by the processing circuitry in the particular time period prior to the first context switch for determining whether to preload said data into said cache memory in response to said second context switch, to retrieve said data, based upon said determination, from said computer memory in response to said second context switch command, and to store said retrieved data in said cache memory based upon said indicator.
(Emphasis added).

Applicant respectfully asserts that *Jensen* in combination with *Burton* and FOLDOC is inadequate to suggest at least the features of claim 1 highlighted hereinabove.

In this regard, *Jensen* appears to teach a system that "[s]aves microprocessor current cache(s)" in response to a "context or partition switch." However, it appears that when restoring the cache, all the data stored is written into "next cache," i.e., the system uses "dual alternating caches." See *Jensen*, column 4, line 62 through column 5, line13. Furthermore, *Jensen* specifically states that ***"[r]estoring the entire cache*** during the context or partition switch may be more efficient than reading individual lines of cache data from memory as the process or virtual machine begins to execute." See *Jensen*, column 6, line 65 through column 7, line 15. (Emphasis added).

Thus, *Jensen* does not appear to teach a system that "store[s] an indicator indicative of whether said data was accessed by the processing circuitry during a particular time period prior to said first context switch," "make[s] a determination, based on said indicator, whether said data was accessed by the processing circuitry in the particular time period prior to the first context switch for determining whether to preload said data into said cache memory in

response to said second context switch," and "retrieve[s] said data, based upon said determination, from said computer memory in response to said second context switch command," as claimed in claim 1.

Furthermore, the Office Action states that *Burton* teaches "an indicator indicative of how often a portion of said cache memory is accessed in executing said instructions from said one program." See Office Action at Page 6. However, the cited art fails to suggest using such an indicator to determine whether data is to be preloaded into a cache, as described in claim 1.

For at least the reason set forth hereinabove, Applicants respectfully assert that the cited art fails to suggest each feature of claim 1, and the 35 U.S.C. §103 rejection of claim 1 should, therefore, be withdrawn.

Claims 2 and 4-6

Claims 2 and 4-6 presently stand rejected under 35 U.S.C. §103 as unpatentable over *Jensen.* in view of *Burton* and FOLDOC. Applicant submits that the pending dependent claims 2 and 4-6 contain all features of their respective independent claim 1. Since claim 1 should be allowed, as argued hereinabove, pending dependent claims 2 and 4-6 should be allowed as a matter of law for at least this reason. *In re Fine*, 5 U.S.P.Q.2d 1596, 1600 (Fed. Cir. 1988). Furthermore, these dependent claims recite patentably distinct features and/or combinations of features that make them allowable, notwithstanding the allowability of their base claim 1.

Claim 7

Claim 7 presently stands rejected under 35 U.S.C. §103 as unpatentable over *Jensen* in view of *Burton* and FOLDOC. Amended claim 7 reads as follows:

> 7. A computer system for efficiently executing instructions of computer programs, comprising:
>
> processing circuitry having a pipeline, said pipeline configured to execute instructions from one of a plurality of programs, said processing circuitry configured to stop executing said one program during a first context switch in response to a first context switch command and to resume executing said one program during a second context switch in response to a second context switch command;
>
> cache memory;
>
> computer memory having a plurality of addresses; and
>
> ***memory control circuitry coupled to said processing circuitry, said memory control circuitry configured to maintain a plurality of mappings, said mappings respectively correlating at least one data value previously written by said pipeline during execution of an instruction and stored in said cache memory with said memory addresses of said computer memory, said memory control circuitry configured to store in said computer memory said mappings and information indicating whether said at least one data value was recently accessed and to make a determination, based on said information, whether said information indicates that said at least one data value was recently accessed prior to the first context switch, said memory control circuitry further configured to preload, into said cache memory, said at least on e data value based on said determination if said information indicates that said at least one data value was recently accessed prior to said first context switch.***
> (Emphasis added).

Applicant respectfully asserts at least those features of claim 7 highlighted above, and the 35 U.S.C. §103 rejection of claim 7 should, therefore, be withdrawn.

In this regard, as argued hereinabove with respect to claim 1, *Jensen* appears to teach a system that "[s]aves microprocessor current cache(s)" in response to a "context or partition switch," However, it appears that when restoring the cache, all the data stored is written into "next cache," i.e., the system uses "dual alternating caches." See *Jensen*, column 4, line 62 through column 5, line13. Furthermore, *Jensen* specifically states that ***"[r]estoring the entire cache*** during the context or partition switch may be more efficient than reading individual

lines of cache data from memory as the process or virtual machine begins to execute." See Jensen, column 6, line 65 through column 7, line 15. (Emphasis added).

However, it does not appear that *Jensen* teaches a system that "store[s] in said computer memory said mappings and information indicating whether said at least one data value was recently accessed," "make[s] a determination, based on said information, whether said information indicates that said at least one data value was recently accessed prior to the first context switch," and "preload[s], into said cache memory, said at least on e data value based on said determination if said information indicates that said at least one data value was recently accessed prior to said first context switch," as claimed in claim 7.

Furthermore, the Office Action states that *Burton* teaches "an indicator indicative of how often a portion of said cache memory is accessed in executing said instructions from said one program." See Office Action at Page 6. However, the cited art fails to suggest using such an indicator to determine whether data is to be preloaded into a cache, as described in claim 1.

For at least the reason set forth hereinabove, Applicants respectfully assert that the cited art fails to suggest each feature of claim 7, and the 35 U.S.C. §103 rejection of claim 7 should, therefore, be withdrawn.

Claims 8, 10 and 11

Claims 8, 10 and 11 presently stand rejected under 35 U.S.C. §103 as unpatentable over *Jensen* in view of *Burton* and FOLDOC. Applicant submits that the pending dependent claims 8, 10 and 11 contain all features of their respective independent claim 7. Since claim 7 should be allowed, as argued hereinabove, pending dependent claims 8, 10 and 11 should be allowed as a matter of law for at least this reason. *In re Fine*, 5 U.S.P.Q.2d 1596, 1600 (Fed. Cir. 1988). Furthermore, these dependent claims recite patentably distinct features and/or combinations of features that make them allowable, notwithstanding the allowability of their base claim 7.

Claim 12

Claim 12 presently stands rejected under 35 U.S.C. §103 as unpatentable over *Jensen* in view of *Burton*. Claim 12 reads as follows:

> 12. A method for efficiently executing instructions of computer programs, comprising the steps of:
> executing a plurality of computer programs in an interleaved fashion;
> switching which of said computer programs is being executed in said executing step;
> storing, prior to said switching step, at an address in computer memory a data value previously written by a pipeline to a cache line in execution of an instruction corresponding to one of said computer programs in said executing step and an indicator indicating if the cache line was recently accessed ;
> identifying said address in response to said switching step;
> ***determining, based on said indicator, whether the data value was recently accessed prior to said switching step;***
> ***retrieving, based on said determining step, said data value from said address based on said identifying step and in response to said switching step if said indicator indicates that the data was recently accessed;*** and
> storing said retrieved data value in cache memory. (Emphasis added).

For at least the reason set forth hereinabove with reference to claim 1, Applicants respectfully assert that the cited art fails to suggest each feature of claim 12, and the 35 U.S.C. §103 rejection of claim 12 should, therefore, be withdrawn.

Claim 13-15

Claims 13-15 presently stand rejected under 35 U.S.C. §103 as unpatentable over *Jensen* in view of *Burton*. Applicant submits that the pending dependent claims 13-15 contain all features of its respective independent claim 12. Since claim 12 should be allowed, as argued hereinabove, pending dependent claims 13-15 should be allowed as a matter of law for at least this reason. *In re Fine*, 5 U.S.P.Q.2d 1596, 1600 (Fed. Cir. 1988). Furthermore,

these dependent claims recite patentably distinct features and/or combinations of features that make them allowable, notwithstanding the allowability of their base claim 12.

Claim 16

Claim 16 presently stands rejected under 35 U.S.C. §103 as unpatentable over *Jensen* in view of *Burton* and FOLDOC. Amended claim 16 reads as follows:

> 16. A method for efficiently executing instructions of computer programs, comprising the steps of:
> executing instructions from a computer program;
> halting said executing step during a first context switch in response to a first context switch command;
> resuming said executing step during a second context switch in response to a second context switch command;
> maintaining a plurality of mappings;
> correlating, via said mappings, data values previously written by a pipeline during the executing step and stored in a cache memory with memory addresses of computer memory outside of said cache memory;
> storing said mappings in said computer memory in response to said first context switch command and information indicative of whether said data values [[]]were accessed during a particular time period prior to said first context switch;
> ***selecting, based on said information and for preloading into the cache memory in response to the second context switch command, at least one data value from at least one of said addresses identified by said mappings; and***
> ***retrieving, based on said mappings and said selecting step, said at least one data value in response to said second context switch command if said information indicates that said at least one data value was accessed during said particular time period***; and
> storing said at least one retrieved data value in said cache memory. (Emphasis added).

Applicant respectfully asserts that *Jensen* in combination with *Burton* and FOLDOC is inadequate to suggest at least the features of claim 16 highlighted hereinabove.

As argued with reference to claim 1, *Jensen* appears to teach a system that "[s]aves microprocessor current cache(s)" in response to a "context or partition switch." However, it appears that when restoring the cache, all the data stored is written into "next cache," i.e., the system uses "dual alternating caches." See *Jensen*, column 4, line 62 through column 5,

line13. Furthermore, *Jensen* specifically states that ***"[r]estoring the entire cache*** during the context or partition switch may be more efficient than reading individual lines of cache data from memory as the process or virtual machine begins to execute." See Jensen, column 6, line 65 through column 7, line 15. (Emphasis added).

Thus, Jensen does not appear to teach a system that "select[s], based on said information and for preloading into the cache memory in response to the second context switch command, at least one data value from at least one of said addresses identified by said mappings" and "retriev[es], based on said mappings and said selecting step, said at least one data value in response to said second context switch command if said information indicates that said at least one data value was accessed during said particular time period," as claimed in claim 16.

Furthermore, the Office Action states that *Burton* teaches "an indicator indicative of how often a portion of said cache memory is accessed in executing said instructions from said one program." See Office Action at Page 6. However, the cited art fails to suggest using such an indicator to determine whether data is to be preloaded into a cache, as described in claim 1.

For at least the reason set forth hereinabove, Applicants respectfully assert that the cited art fails to suggest each feature of claim 16, and the 35 U.S.C. §103 rejection of claim 16 should, therefore, be withdrawn.

Claims 18 and 19

Claims 18 and 19 presently stand rejected under 35 U.S.C. §103 as unpatentable over *Jensen* in view of *Burton*. Applicant submits that the pending dependent claims 18 and 19 contain all features of their respective independent claim 16. Since claim 16 should be allowed, as argued hereinabove, pending dependent claims 18 and 19 should be allowed as a matter of law for at least this reason. *In re Fine*, 5 U.S.P.Q.2d 1596, 1600 (Fed. Cir. 1988). Furthermore, these dependent claims recite patentably distinct features and/or combinations of features that make them allowable, notwithstanding the allowability of their base claim 16.

Claim 20

Claim 20 presently stands rejected under 35 U.S.C. §103 as unpatentable over *Jensen* in view of *Burton* and FOLDOC. Amended claim 20 reads as follows:

> 20. A computer system for efficiently executing instructions of computer programs, comprising:
> computer memory; and
> a processing unit comprising cache memory and ***logic configured to store in said computer memory a value indicative of whether a portion of said cache memory was recently accessed by said processor and a mapping associated with said value, said mapping indicative of a location in said computer memory storing data previously requested or previously written by an instruction of a first process being executed by the processing unit when the processing unit context switches out the first process for processing of a second process, the logic further configured to select said data for preloading in said cache memory after said second context switch if said value indicates that the data was recently accessed, retrieve said data based on said value and store said data in said cache if said value indicates that the data was recently accessed, said processing unit continuing execution of said first process with the retrieved data when the processing unit context switches out the second process and context switches in the first process.*** (Emphasis added).

For at least the reasons similar to those set forth hereinabove with reference to claim 16, Applicants respectfully assert that the cited art fails to suggest each feature of claim 20, and the 35 U.S.C. §103 rejection of claim 20 should, therefore, be withdrawn.

Claim 21

Claim 21 presently stands rejected under 35 U.S.C. §103 as unpatentable over *Jensen* in view of *Burton*. Amended claim 21 reads as follows:

> 21. A method for efficiently executing instructions of computer programs, comprising the steps of:
> storing a value indicative of whether data in cache memory was recently accessed by a processing unit;
> storing in said memory a mapping corresponding to said value, said mapping indicative of a location in computer memory storing said data when the processing unit context switches out the first process for processing of a second process;
> ***determining whether said data was recently accessed prior to said processing unit switching out the first process;***
> ***preloading, based on said determining step, said data in said cache for execution of said first process by said processing unit if said value indicates that the data was recently accessed in said determining step.*** (Emphasis added).

For at least the reasons similar to those set forth hereinabove with reference to claim 1, Applicants respectfully assert that the cited art fails to suggest each feature of claim 21, and the 35 U.S.C. §103 rejection of claim 21 should, therefore, be withdrawn.

Claims 22-27

Claims 22-27 presently stand rejected under 35 U.S.C. §103 as unpatentable over *Jensen* in view of *Burton*. Applicant submits that the pending dependent claims 22 through 27 contain all features of their respective independent claim 21. Since claim 21 should be allowed, as argued hereinabove, pending dependent claims 22 through 27 should be allowed as a matter of law for at least this reason. *In re Fine*, 5 U.S.P.Q.2d 1596, 1600 (Fed. Cir. 1988). Furthermore, these dependent claims recite patentably distinct features and/or combinations of features that make them allowable, notwithstanding the allowability of their base claim 21.

Claim 30

Claim 30 presently stands rejected under 35 U.S.C. §103 as unpatentable over *Jensen* in view of *Burton*. Amended claim 30 reads as follows:

> 30. A system, comprising:
> computer memory; and
> a processing unit comprising cache memory, said cache memory comprising a plurality of cache lines storing data written or read by a first process during execution by said processing unit, ***said processing unit further comprising logic configured to periodically assert one of a plurality of flags associated with each of said cache lines when said flag's associated cache line is accessed, said logic further configured to store in said computer memory said flags and data corresponding to the associated cache lines accessed by said first process upon a first context switch, said logic further configured to select data for preloading into the cache memory based upon the asserted flags in response to a second context switch*** . (Emphasis added).

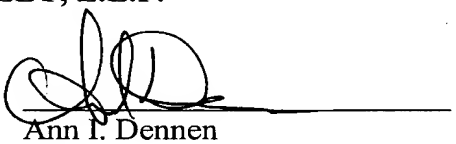
For at least the reasons similar to those set forth hereinabove with reference to claim 16, Applicants respectfully assert that the cited art fails to suggest each feature of claim 30, and the 35 U.S.C. §103 rejection of claim 30 should, therefore, be withdrawn.

CONCLUSION

Applicant respectfully requests that all outstanding objections and rejections be withdrawn and that this application and all presently pending claims be allowed to issue. If the Examiner has any questions or comments regarding Applicant's response, the Examiner is encouraged to telephone Applicant's undersigned counsel.

Respectfully submitted,

THOMAS, KAYDEN, HORSTEMEYER & RISLEY, L.L.P.

By: 

Ann I. Dennen
Reg. No. 44,651
(256) 704-3900

Hewlett-Packard Development Company, L.P.
Intellectual Property Administration
P.O. Box 272400
Fort Collins, Colorado 80527-2400